

## **DETAILED ACTION**

This action is in response to RCE filing of 7 December 2007. Claims 1, 25, 27, 28, 34 and 38 have been amended. Claims 6, 10, 11 and 24 has been canceled. Claims 1-5, 7-9, 12-23 and 25-38 are pending and have been considered below.

### ***Response to Arguments***

1. Applicant's arguments with respect to claims 1-5, 7-9, 12-23 and 25-38 have been considered but are moot in view of the new ground(s) of rejection.

### ***Claim Objections***

2. Claims 7 and 25 are objected to because of the following informalities:

Claim 7: Claim 7 depends from claim 6, however, claim 6 has been canceled.

Claim 25: In line 7 of claim 25, the word "directly" between the words "hierarchical" and "location" seems to be a misspelling of the word "directory".

Appropriate correction is required.

### ***Claim Rejections - 35 USC § 101***

3. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 1-5, 7-9, 12-23 and 25-27 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. The system claimed in the claims appears to consist of software per se. Software per se is none of a process, machine, manufacture or composition and is therefore, a non-statutory category of invention.

***Claim Rejections - 35 USC § 103***

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

6. Claims 1-5, 12-15, 20-23, 25-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over of Cowart (Mastering Windows 95) in view of Ortega et al. (US 6,489,968 B1).

Claim 1: Cowart discloses a system for data presentation, comprising:

- a. A sorting component to determine categories relating to one or more items for display (page 178, items are sorted according to categories identified by file extensions), and
- b. A cluster component that groups the categories according to discretized states in order to control visible output to the display (page 499, items can be clustered in a folder), wherein
- c. The states include at least a packed state that caused items in the grouped categories to be displayed as a single icon when viewed from outside of the grouped categories (pages 197-198, pages 508 and 529, Short-cuts of items from any folder can be placed in a folder. When these items are viewed from outside the group, the group appears as a folder.), and
- d. An unpacked state that causes each item in the grouped categories to be displayed as an individual icon when viewed from outside of the grouped categories (pages 197-198, pages 508 and 529, Short-cuts of items from any folder can be placed in a folder. When these items are viewed from outside the group, the group appears as a folder. When the folder is opened, the items are viewed individually),

but does not disclose

- a. the state is assigned as a property to each grouped category,
- b. items in the grouped categories are displayed as a single icon when viewed from any higher level hierarchical folder outside of the grouped categories, and
- c. each item in the grouped categories to be displayed as an individual icon when viewed from any higher level hierarchical folder outside of the grouped categories.

However, Ortega discloses lower level item nodes and category nodes in a hierarchy are elevated for display at higher levels of the hierarchy. These items and categories are assigned a state, such as popularity, to determine which ones get elevated (column 4, lines 16-40, column 6, lines 5-20, column 7, lines 32-50, figures 1a and 1b).

Therefore, it would have been obvious to having ordinary skill in the art at the time of the invention to add

- a. the state is assigned as a property to each grouped category,
- b. items in the grouped categories are displayed as a single icon when viewed from any higher level hierarchical folder outside of the grouped categories, and
- c. each item in the grouped categories to be displayed as an individual icon when viewed from any higher level hierarchical folder outside of the grouped categories

to Cowart. One could have been motivated to add

- a. the state is assigned as a property to each grouped category,

- b. items in the grouped categories are displayed as a single icon when viewed from any higher level hierarchical folder outside of the grouped categories, and
- c. each item in the grouped categories to be displayed as an individual icon when viewed from any higher level hierarchical folder outside of the grouped categories

to Cowart in order to call to attention of the users the nodes (items and/or categories) of a hierarchical structure that are most popular or important. (Ortega: column 1, lines 5-10).

Claim 2: Cowart and Ortega disclose a system as in claim 1 above and Cowart further discloses the user interface that is used to view and manipulate files and folders which includes access to hard drives and floppy drives (page 498, figure 12.2).

Claim 3: Cowart and Ortega disclose a system as in claim 1 above and Cowart further discloses files and folders can be viewed and manipulated by a user interface (page 498, figure 12.2).

Claim 4: Cowart and Ortega disclose system as in claim 2 above and Cowart further discloses a tree view as part of the user interface to view and manipulate files and folders (page 498, figure 12.2).

Claim 5: Cowart and Ortega disclose the system as in claim 2, and Cowart further discloses the cluster component controls content merging of subordinate and sibling nodes at the user interface (page 498, figure 12.2).

Claim 12: Cowart and Ortega disclose a system as in claim 1 above and Cowart further discloses setting the rules to determine how items are to be displayed (page 235, Sorting and Tidying Up The Listing).

Claim 13: Cowart and Ortega disclose a system as in claim 1 above and Cowart further discloses setting the state of a file (page 525, figure 12.9).

Claim 14: Cowart and Ortega disclose a system as in 13 above and Cowart further discloses setting the state of a group (subset) of items (page 525, figure 12.9)

Claim 15: Cowart and Ortega disclose a system as in claim 13 and Cowart further discloses setting a flag indicating the state of an item (page 525, figure 12.9) but does not disclose setting a flag for the states of packed and unpacked. It would have been obvious to one having ordinary skill in the art at the time of the invention for Cowart to set a flag indicating the states of packed and unpacked. One would have been motivated to set a flag indicating the states of packed and unpacked to be able to determine when an item is in the packed state or the unpacked state.

Claim 20: Cowart and Ortega disclose a system as in claim 1 above and Cowart further discloses showing the static contents of a directory (page 498, Figure 12.2) and for showing a dynamic query of files and folders (page 36 and 37, The Find Button).

Claim 21: Cowart and Ortega disclose a system as in claim 20 above and Cowart further discloses showing the results of a Find query in a similar fashion as the contents of a directory or the directory itself, thereby associating an unpacked group and a packed group with a dynamic query (page 36 and 37, The Find Button).

Claim 22: Cowart and Ortega disclose a system as in claim 1 above and Cowart further discloses setting the state of an item (page 525, figure 12.9) but does not disclose that the item is set to a default state. It would have been obvious to one having ordinary skill in the art at the time of the invention for Cowart to set the initial state of an item to a default state. One would have been motivated to set the initial state to a default state in order that the item had a known initial state.

Claim 23: Cowart and Ortega disclose a system as in claim 22 above and Cowart further discloses setting the state of an item (page 525, figure 12.9) but does not disclose the rules for setting the default state of an item. It would have been obvious to one having ordinary skill in the art at the time of the invention for Cowart to define rules for setting the default state. One would have been motivated to define rules for setting the default state of an item in order to set the default state of an item to a known state.

Claim 25: Cowart discloses a system for determining a state for a group (subset) of items and for setting the state of a group (subset) of items (page 525, figure 12.9) and for displaying the items according to assigned state (page 498, Figure 12.2), wherein the states include at least a packed state that causes items in the grouped categories to be displayed as a single icon when viewed from outside of the grouped categories (pages 197-198, pages 508 and 529, Short-cuts of items from any folder can be placed in a folder. When these items are viewed from outside the group, the group appears as a folder.), and an unpacked state that causes each item in the grouped categories to be displayed as an individual icon when viewed from outside of the grouped categories (pages 197-198, pages 508 and 529, Short-cuts of items from any folder can be placed in a folder. When these items are viewed from outside the group, the group appears as a folder. When the folder is opened, the items are viewed individually)

but does not disclose

- a. the state is assigned as a property to each grouped subset,
- b. items in the subset are displayed as a single icon when viewed from any higher level hierarchical directory location outside of the subset, and
- c. each item in the subset is displayed as an individual icon when viewed from any higher level directory location outside of the grouped categories.

However, Ortega discloses lower level item nodes and category nodes in a hierarchy are elevated for display at higher levels of the hierarchy. These items and categories are assigned a state, such as popularity, to determine which ones get elevated (column 4, lines 16-40, column 6, lines 5-20, column 7, lines 32-50, figures 1a and 1b).

Therefore, it would have been obvious to having ordinary skill in the art at the time of the invention to add

- a. the state is assigned as a property to each grouped subset,
- b. items in the subset are displayed as a single icon when viewed from any higher level hierarchical directory location outside of the subset, and
- c. each item in the subset is displayed as an individual icon when viewed from any higher level directory location outside of the grouped categories

to Cowart. One could have been motivated to add

- a. the state is assigned as a property to each grouped subset,
- b. items in the subset are displayed as a single icon when viewed from any higher level hierarchical directory location outside of the subset, and
- c. each item in the subset is displayed as an individual icon when viewed from any higher level directory location outside of the grouped categories

to Cowart in order to call to attention of the users the nodes (items and/or categories) of a hierarchical structure that are most popular or important. (Ortega: column 1, lines 5-10).

Claim 26: Cowart and Ortega disclose a system as in claim 25 above and Cowart further discloses displaying items (page 498, figure 12.2) but does not specifically disclose displaying a packed group, an unpacked group, and an overlapping group. It would have been obvious to one having ordinary skill in the art at the time of the invention for Cowart to display packed groups, unpacked groups and overlapping groups. One would have been motivated to display a packed group, an unpacked group, and an overlapping group so they could be viewed by the user.

Claim 27: Cowart discloses a system as in claim 26 above, and further discloses clicking a mouse to control a display (page 104, Note).

7. Claims 28-31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hasegawa (US 6,513,038) in view of Ortega et al. (US 6,489,968 B1).

Claim 28: Hasegawa discloses a system and method for creating a directory (grouping items) called a view directory from a content directory according to class, entry and attribute which is hidden from external view (column 9, lines 66-67 and column 10, lines 1-3) but does not disclose creating a view directory in a packed state or an unpacked state or displaying them in separate views. However, It would have been obvious to one having ordinary skill in the art at the time of the invention to use the state attribute, i.e. packed or unpacked, to define directory nodes and members when creating a view directory from a content directory and to display them in separate views. One would

have been motivated to define directory nodes and members in a packed or unpacked state and to display them in separate views in order to provide the user with additional information about these nodes and to maintain each original view for later viewing, but does not disclose

- a. items in the group are displayed as a single item when viewed from any higher level hierarchical structural location outside of the grouped categories, and
- b. each item in the group is displayed as an individual when viewed from any higher level hierarchical structural location outside of the grouped categories.

However, Ortega discloses lower level item nodes and category nodes in a hierarchy are elevated for display at higher levels of the hierarchy. These items and categories are assigned a state, such as popularity, to determine which ones get elevated (column 4, lines 16-40, column 6, lines 5-20, column 7, lines 32-50, figures 1a and 1b).

Therefore, it would have been obvious to having ordinary skill in the art at the time of the invention to add

- a. items in the group are displayed as a single item when viewed from any higher level hierarchical structural location outside of the grouped categories, and
- b. each item in the group is displayed as an individual when viewed from any higher level hierarchical structural location outside of the grouped categories

to Hasegawa. One could have been motivated to add

- a. items in the group are displayed as a single item when viewed from any higher level hierarchical structural location outside of the grouped categories, and
- b. each item in the group is displayed as an individual when viewed from any higher level hierarchical structural location outside of the grouped categories

to Hasegawa in order to call to attention of the users the nodes (items and/or categories) of a hierarchical structure that are most popular or important. (Ortega: column 1, lines 5-10).

Claim 29: Hasegawa and Ortega disclose a method as in 28 above and Hasegawa further discloses creating a directory called a view directory from a content directory according to class, entry and attribute (column 9, lines 66-67 and column 10, lines 1-3).

Claim 30: Hasegawa and Ortega disclose a system as in claim 9 above and a method for accessing a data management directory (Hasegawa, Abstract, lines 1, 2) and for creating a view directory from a content directory based on class, entry and attribute (Hasegawa, column 9, lines 66-67 and column 10, lines 1-3) but does not disclose persisting states to data storage. It would have been obvious to one having ordinary skill in the art at the time of the invention to persist the classes, entries and attributes to data storage. One would have been motivated to persist the classes, entries and attributes to data storage in order to avoid inputting this data in each time a view directory is to be created.

Claim 31: Hasegawa and Ortega disclose a method as in claim 29 above and Hasegawa further discloses creating a new directory containing the intersection, or overlap, of a plurality of directors (column 10, lines 19-21, column 20, lines 34-67, column 21, lines 1-7).

8. Claims 7, 8, 16-19, 34-38 are rejected under 35 U.S.C. 103(a) as being unpatentable over Cowart (Mastering Windows 95) in view of Ortega et al. (US 6,489,968 B1) and further in view of Hasegawa (US 6,513,038).

Claim 7: Cowart and Ortega disclose a system as in claim 6 above and method for accessing a data management directory (Hasegawa, Abstract, lines 1, 2) and for creating a view directory from a content directory based on class, entry and attribute (Hasegawa, column 9, lines 66-67 and column 10, lines 1-3) but do not disclose persisting states to data storage. It would have been obvious to one having ordinary skill in the art at the time of the invention to persist the classes, entries and attributes to data storage. One would have been motivated to persist the classes, entries and attributes to data storage in order to avoid inputting this data in each time a view directory is to be created.

Claim 8: Cowart, Ortega and Hasegawa disclose a system as in claim 7 above and Hasegawa discloses creating a directory called a view directory from a content directory

according to class, entry and attribute (column 9, lines 66-67 and column 10, lines 1-3). Therefore, it would have been obvious to one having ordinary skill in the art at the time of the invention for Cowart to associate attributes representing state with properties of a group. One would have been motivated to associate attributes representing state with properties of a group in order for all members of a group to be identified with the same state.

Claim 16, 18: Cowart and Ortega disclose a computerized interface for data presentation comprising a sorting component and a cluster component as in claim 1 above but does not disclose an overlapping group that includes content from various groups. However, Hasegawa discloses a similar system which further discloses creating a new directory containing the intersection, or overlap, of a plurality of directors (column 10, lines 19-21, column 20, lines 34-67, column 21, lines 1-7). Therefore, it would have been obvious to one having ordinary skill in the art at the time of the invention for Cowart to include overlapping groups that contain content from various groups. One would have been motivated to include overlapping groups in order to filter out items that are common to two groups.

Claim 17: Cowart, Ortega and Hasegawa disclose a system as in claim 16 above including creating a new directory containing the intersecting or overlapping contents of a plurality of directories (Hasegawa, column 10, lines 19-21, column 20, lines 34-67, column 21, lines 1-7) but do not disclose including a recycle group and an archive group

as an overlapping group. However, it would have been obvious to one having ordinary skill in the art at the time of the invention for Cowart to apply any desirable name to an overlapping group. One would have been motivated to apply a name to an overlapping group in order to distinguish it from other overlapping groups.

Claim 19: Cowart and Hasegawa disclose a system as in claim 18 above and Hasegawa further discloses creating a new directory containing the union of two other directors (column 10, lines 10-14, columns 17, 18 and 19). Therefore, it would have been obvious to one having ordinary skill in the art at the time of the invention for Cowart to provide the union of two groups. One would have been motivated to provide the union of two groups so that two groups could be viewed as one group.

Claims 34-37: Cowart discloses a computer readable medium having a data structure stored thereon, comprising:

- a. A packed state that caused items in the grouped categories to be displayed as a single icon when viewed from outside of the grouped categories (pages 197-198, pages 508 and 529, Short-cuts of items from any folder can be placed in a folder. When these items are viewed from outside the group, the group appears as a folder.), and
- b. An unpacked state that causes each item in the grouped categories to be displayed as an individual icon when viewed from outside of the grouped categories (pages 197-198, pages 508 and 529, Short-cuts of items from any

folder can be placed in a folder. When these items are viewed from outside the group, the group appears as a folder. When the folder is opened, the items are viewed individually),

but does not disclose

- a. A first data field relating to at least one group property associated with a subset of data items for display
- b. A second data field for the data items, and
- c. A third data field to control how the data items are to be directed to a computerized display
- d. items in the subset are displayed as a single icon when viewed from any higher level folder outside of the subset, and
- e. each item in the subset is displayed as an individual when viewed from any higher level folder outside of the grouped categories.

However, Hasegawa discloses a data structure including a data field for a group property, a data field for holding data, and a data field for controlling how the data items are to be directed to a display (figures 12A and 12B). Therefore, it would have been obvious to one having ordinary skill in the art at the time of the invention to add a data structure including a data field for a group property, a data field for holding data, and a data field for controlling how the data items are to be directed to a display because Cowart defines and applies properties to both folders (packed state) and files (unpacked state).

Ortega discloses lower level item nodes and category nodes in a hierarchy are elevated for display at higher levels of the hierarchy. These items and categories are assigned a state, such as popularity, to determine which ones get elevated (column 4, lines 16-40, column 6, lines 5-20, column 7, lines 32-50, figures 1a and 1b). Therefore, it would have been obvious to having ordinary skill in the art at the time of the invention to add

- a. items in the subset are displayed as a single icon when viewed from any higher level folder outside of the subset, and
- b. each item in the subset is displayed as an individual when viewed from any higher level folder outside of the grouped categories

to Cowart and Hasegawa. One could have been motivated to add

- a. items in the subset are displayed as a single icon when viewed from any higher level folder outside of the subset, and
- b. each item in the subset is displayed as an individual when viewed from any higher level folder outside of the grouped categories

to Cowart and Hasegawa in order to call to attention of the users the nodes (items and/or categories) of a hierarchical structure that are most popular or important. (Ortega: column 1, lines 5-10).

Claim 38: Cowart and Hasegawa disclose a data structure as in 37 above but does not disclose associating the state with a packed state, and unpacked state, an overlapping state, and a dynamic state. However, it would have been obvious to one having ordinary skill in the art at the time of the invention for Hasegawa to associate the state with all known or possible states, such as packed, unpacked, overlapping, and dynamic states. One would have been motivated to associate the state with a packed state, and unpacked state, an overlapping state, and a dynamic state in order to be able to distinguish one state from another.

9. Claim 9 is rejected under 35 U.S.C. 103(a) as being unpatentable over Cowart (Mastering Windows 95) in view of Hasegawa (US 6,513,038) and Ortega et al. (US 6,489,968 B1) and further in view of Newman (US 2004/0139231 A1).

Claim 9: Cowart, Hasegawa and Ortega disclose a system as in claim 8 above for creating a directory called a view directory from a content directory according to class, entry and attribute (column 9, lines 66-67 and column 10, lines 1-3) but does not disclose associating property data with meta data. Newman discloses a similar system in which meta data is associated with property data (paragraph 0068, lines 1-3). Therefore, it would have been obvious to one having ordinary skill in the art at the time of the invention for Cowart to associate property data with meta data. One would have been motivated to associate property data with meta data in order to be able to view in human readable format what properties are applied to an item.

10. Claims 32 and 33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hasegawa(US 6,513,038) in view of Ortega et al. (US 6,489,968 B1) and futher in view of Cowart (Mastering Windows 95).

Claim 32: Hasegawa, Ortega and Cowart disclose a method for creating a directory called a view directory from a content directory (Hasegawa, column 9, lines 66-67, column 10, lines 1-3) and displaying an icon for files and folders (Cowart, page 498, figure 12.2) as in claim 31 above but does not disclose displaying an icon for a packed, an unpacked group or an overlapping group. It would have been obvious to one having ordinary skill in the art at the time of the invention for Cowart to display an icon for a packed, an unpacked group or an overlapping group. One would have been motivated to display an icon for a packed, an unpacked group or an overlapping group in order to distinguish each group from other groups.

Claim 33: Hasegawa, Ortega and Cowart disclose a method as in claim 32 above and Hasegawa further discloses creating a directory called a view directory from a content directory according to class, entry and attribute (column 9, lines 66-67 and column 10, lines 1-3). The contents of the view directory could be viewed by opening the directory.

***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to John M. Heffington whose telephone number is (571) 270-1696. The examiner can normally be reached on Mon - Fri 8:00 - 5:30 EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Weilun Lo can be reached on (571) 272-4847. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Application/Control Number: 10/758,743  
Art Unit: 2179

Page 22

/Ba Huynh/

Primary Examiner, Art Unit 2179